Amendments to the Claims:

This listing of claims will replace all prior versions, and listing, of claims in the application.

1. – 13. (Canceled)

14. (Currently Amended):   A method for replicating data over a network including a master server and multiple slave servers, the method comprising the steps of:

without being requested by the slave server, sending a packet of information from a master server to each slave server on the network, the information relating to a change in data stored on the master server and including an update for slave servers and containing a ~~current~~ version number ~~for a present state of the data, the information further relating to previous changes in the data and a version number for each previous change~~;

thereafter, allowing each slave server to use the version number to determine whether the slave server has been updated to correspond to the current version ~~number~~;

thereafter, allowing each slave server to commit the update in the information if the slave server has not missed a previous change; and

allowing each slave server having missed a previous change thus unable to use the update to request that previous change be sent from the master server to the slave server before the slave server commits the packet of information.

15. (Original): A method according to claim 14, further comprising:

committing the packet of information to a slave server.

16. (Original): A method according to claim 14, further comprising:

aborting the commit of the packet of information if a slave server cannot commit the update.

17. (Original): A method according to claim 14, further comprising:

determining the scope of the delta before sending it from the master server.

18. (Original): A method according to claim 14, further comprising:

including the scope of each the previous changes in the delta.

19. (Currently Amended):   A method for replicating data over a network including a master server and multiple slave servers, the method comprising the steps of:

without being requested by the slave server, sending a packet of information from a master server to each slave server on the network, the information relating to a change in the data stored on the master server and including an update for slave servers and containing a ~~prior~~ version number ~~for a prior state and a Previously Presented version number for a Previously Presented state of the data, the information further relating to previous changes in the data and a previous version number for each previous change~~;

thereafter, allowing each slave server to use the version number to determine whether ~~the data on~~ the slave server can use the update ~~corresponds to the prior version number contained in the packet~~;

thereafter, allowing each slave server to commit the update in the packet of information if they are determined to be able to update ~~the data on the slave server corresponds to the prior version number contained in the packet, the commit also updating the version of the slave server to the Previously Presented version number~~; and

allowing each slave server that was determined to be unable to update ~~not corresponding to the prior version number~~ to request that a delta be sent from the master server ~~containing the information necessary to update the slave to the prior version number before the slave server commits the packet of information~~.

20. (Currently Amended):   A method for replicating data over a network including a master server and multiple slave servers, the method comprising the steps of:

without being requested by the slave server, sending a packet of information from a master server to each slave server on the network, the information relating to a change in the data stored on the master server and containing a version number and including an update for slave servers ~~for the prior state and a version number for the Previously Presented state of the data, the information further relating to previous changes in the data and a version number for each previous change~~;

thereafter, allowing each slave server to determine <u>using the version number</u> whether <u>the slave servers can be updated with the update</u> ~~a data on the slave server corresponds to the prior version number contained in the packet~~;

thereafter, allowing each slave server to commit <u>update in</u> the packet of information if <u>it has been determined that the slave server can update</u> ~~the data on the slave server corresponds to a prior version number contained in the packet,~~ the commit also updating the version of the slave server ~~to the Previously Presented version number~~; and

allowing each slave server <u>which has been determined to not be updateable with the update</u> ~~not corresponding to the prior version number~~ to request that a delta be sent from the master server containing the information necessary ~~to update the slave to the Previously Presented version number~~.

21. (Currently Amended):   A method for replicating data from a master server to multiple slave servers over a network, the method comprising the steps of:

sending a packet of information from the master server to a slave server, the information relating to a change in a data stored on the master server <u>and including an update for slave servers</u> and containing a version number ~~for a present state of the data~~;

thereafter, receiving the packet of information to a slave server;

thereafter, allowing the slave server <u>to use the version number to</u> determine whether the slave server has been updated to correspond to [[the]] <u>a current</u> version number ~~contained in the packet~~, and to further determine whether the slave server can process the <u>update in the</u> packet of information if needed to update to correspond to the version number contained in the packet;

thereafter, sending a signal from the slave server to the master server, the signal indicating whether the slave server needs to be updated and whether the slave server can process the update; and

determining whether each of the slave servers can commit the update,

thereafter, sending a response signal from the master server to the slave server indicating whether the slave servers should commit to the information contained in the packet; and

thereafter, committing the <u>update in the</u> packet of information to the slave servers if so indicated by the response signal<u>;</u>

wherein the update is committed to all the slaves if all the slaves are able to commit the update and the update is not committed to any slave, if any slave is not able to commit the update.

22. (Canceled)

23. (Previously Presented):   A method according to claim 21, further comprising:
determining whether each of the multiple slave servers has sent a response back to the master server.

24. (Previously Presented):   A method according to claim 21, further comprising:
determining whether any of the multiple slave servers can commit the data.

25. (Canceled)

26. (Previously Presented):   A method according to claim 21, further comprising:
aborting the update only if any of the multiple slave servers cannot process the commit.

27. (Canceled)

28. (Previously Presented):   A method according to claim 21, further comprising:
multicasting an update to any of the multiple slave servers that were not able to process the commit.

29. (Original): A method according to claim 21, further comprising:
heartbeating the Previously Presented version number to any of the at least one slave server that were not able to process the commit.

30. (Original): A method according to claim 21, further comprising:
requesting a delta be sent to a slave server that was not able to process the commit.

31. (Currently Amended): A method for replicating data over a network, the method comprising the steps of:

(a) determining whether the replication should be accomplished in a one or two phase method;

(b) if the replication is determined to be accomplished in a one phase method, sending replication information determined to be accomplished in a one phase method by:

sending a packet of information from the master server to the slave server, the information relating to a change in data stored on the master server and containing a version number for a present state of the data;

thereafter, receiving the packet of information to a slave server;

thereafter, allowing the slave server to determine whether the data on the slave server has been updated to correspond to the version number; and

thereafter, requesting a delta be sent from the master server to the slave server if the slave server does not correspond to the version number, the delta containing information needed to update the slave server;

(c) if the replication is determined to be accomplished in a two phase method, sending replication information determined to be accomplished in a two phase method by:

sending a packet of information from the master server to the slave server, the information relating to a change in the data stored on the master server and containing a version number for the present state of the data;

thereafter, allowing the slave server to determine whether the slave server has been updated to correspond to the version number, and to further determine whether the slave server can process the packet of information;

thereafter, sending a signal from the slave server to the master server indicating whether the slave server needs to be updated and whether the slave server can process the packet of information;

thereafter, sending a response signal from the master server to the slave server indicating whether the slave server should commit to the packet of information; and

thereafter, committing the packet of information to the slave server if so indicated by the response signal;

wherein the update is committed to all the slaves if all the slaves are able to commit the update and the update is not committed to any slave, if any slave is not able to commit the update.

32. (Currently Amended):   A method for replicating data over a network, the method comprising the steps of:

(a) determining whether replication should be accomplished in a one or two phase method;

(b) if the replication is determined to be accomplished in a one phase method, sending data to be replicated in a one phase method by:

sending a version number for a current state of the data from a master server to a slave server;

thereafter, requesting a delta be sent from the master server to the slave server if the data on the slave server does not correspond to the version number; and

(c) if the replication is determined to be accomplished in a two phase method, sending data to be replicated in a two phase method by:

sending a packet of information from the master server to a slave server;

thereafter, determining whether the slave server can process the packet of information; and

thereafter, committing the packet of information to the slave server if the slave server can process the packet of information;

wherein the update is committed to all the slaves if all the slaves are able to commit the update and the update is not committed to any slave, if any slave is not able to commit the update.

33. (Currently Amended):   A method for replicating data from a master to a plurality of slaves on a network, the method comprising the steps of:

(a) determining whether replication should be accomplished in a one or two phase method;

(b) if the replication is determined to be accomplished in a one phase method, sending data to be replicated in a one phase method by:

sending a version number for a current state of the data from the master to each slave; and

thereafter, requesting a delta be sent from the master to each slave containing data that does not correspond to the version number;

(c) if the replication is determined to be accomplished in a two phase method, sending data to be replicated in a two phase method by:

sending a packet of information from the master to each slave; and

thereafter, committing the packet of information to the slaves if each of the plurality of slaves can process the packet of information;

wherein the update is committed to all the slaves if all the slaves are able to commit the update and the update is not committed to any slave, if any slave is not able to commit the update.

34. (Currently Amended):   A method for replicating data from a master to a plurality of slaves on a network using one and two phase methods, the method comprising the steps of:

(a) sending data to be replicated in a one phase method by sending a version number for a current state of the data from the master to each slave so that, thereafter, each slave may request a delta to be sent from the master to the slave to update the data on the slave; and

(b) sending data to be replicated in a two phase method by sending a packet of information from the master to each slave, thereafter, the packet of information to be committed by each slave if every slave is able to commit the packet of information;

wherein the update is committed to all the slaves if all the slaves are able to commit the update and the update is not committed to any slave, if any slave is not able to commit the update.

35. (Currently Amended):   A method for replicating data on a clustered network using one and two phase methods, each network cluster containing a cluster master and at least one cluster slave, the method comprising the steps of:

(a) sending data to be replicated in a one phase method by sending a version number for the current state of the data from a first cluster master to all other cluster masters so thereafter the other cluster masters may each request a delta; and

(b) sending data to be replicated in a two phase method by sending a packet of information from the first cluster master to each other cluster master, thereafter the packet of information to be committed by the other cluster masters if the other cluster masters are able to commit the packet of information;

wherein the update is committed to all the slaves if all the slaves are able to commit the update and the update is not committed to any slave, if any slave is not able to commit the update.

36. (Original): A method according to claim 35, further comprising:

sending the data from each cluster master to each cluster slave in the cluster with that cluster master by a one-phase method.

37. (Previously presented): A method according to claim 35, further comprising:

sending the data from each cluster master to each cluster slave in the cluster with that cluster master by a two-phase method.

38. – 41. (Canceled)

42. (Currently Amended):   A method comprising:

at a slave server, without being requested by the slave server, receiving a first delta update from a master server, the first delta update being adapted to be used to update data from a first version to a current version;

at the slave server, checking whether data at the slave server is the first version;

if so, updating the data at the slave server with the first delta update;

if not, requesting a second delta update from the master server, the second delta update being adapted to update from a second version to the current version;

wherein when the second delta update is requested, the slaver server receives the second delta update from the master server and uses the second delta update to update the data at the slave to the current version.

43. (Previously Presented):   The method of claim 42, further comprising:

storing an original copy of the data on the master server.

44. (Previously Presented):   The method of claim 42, further comprising:
        persistently caching the data on a local disk for each slave server.

45. (Previously Presented):   The method of claim 42, further comprising:
        determining a unique version number for the current state of the data on the master server if the data has changed.

46. (Previously Presented):   The method of claim 42, wherein if the first delta update is to be committed, the first delta update is committed in a transactional manner.

47. (Previously Presented):   The method of claim 46, wherein the slave server sends a message to the master server when it is ready to commit the first delta update.

48. (Previously Presented):   The method of claim 47, wherein when all of the slave servers are ready to commit the first delta update, the master server sends a commit message to all of the slave servers.

49. (Currently Amended):   A computer readable storage medium containing code to:
        at a slave server, receive a first delta update from a master server;
the first delta update being adapted to be used to update data from a first version to a current version;
        at the slave server, check whether data at the slave server is the first version,
        if so, update the data at the slave server with the first delta update;
        if not, requesting a second delta update from the master server, the second delta update being adapted to update from a second version to the current version;
        wherein when the second delta update is requested, the slaver server receives the second delta update from the master server and uses the second delta update to update the data at the slave to the current version.

50. (Currently Amended):   A computer readable <u>storage</u> medium of claim 49 containing code to, further comprising:

storing an original copy of the data on the master server.

51. (Currently Amended):   A computer readable <u>storage</u> medium of claim 49 containing code, further comprising:

persistently caching the data on a local disk for each slave server.

52. (Currently Amended):   A computer readable <u>storage</u> medium of claim 49 containing code, further comprising:

determining a unique version number for the current state of the data on the master server if the data has changed.

53. (Currently Amended):   A computer readable <u>storage</u> medium of claim 49 containing code, wherein if the first delta update is to be committed, the first delta update is committed in a transactional manner.

54. (Currently Amended):   A computer readable <u>storage</u> medium of claim 53 containing code, wherein the slave server sends a message to the master server when it is ready to commit the first delta update.

55. (Previously Presented):   A method comprising:

at a master server, sending a first delta update to a slave server, the first delta update adapted to be used to update data from a first version to a current version;

if data at the slave server was at the first version, receiving a message that the slaver server can commit the first delta update;

if not, receiving a message from the slave server that the data at the slave server is at a second version and, at the master server, sending a second delta update adapted to be used to update data from the second version to a current version.

56. (Previously Presented):   The method of claim 55, further comprising:

storing an original copy of the data on the master server.

57. (Previously Presented):   The method of claim 55, further comprising:
persistently caching the data on a local disk for each slave server.

58. (Previously Presented):   The method of claim 55, further comprising:
determining a unique version number for the current state of the data on the master server
if the data has changed.

59. (Previously Presented):   The method of claim 55, wherein if the first delta update is to be
committed, the first delta update is committed in a transactional manner.

60. (Previously Presented):   The method of claim 55, wherein the slave server sends a message
to the master server when it is ready to commit the first delta update.

61. (Previously Presented):   The method of claim 60, wherein when all of the slave servers are
ready to commit the first delta update, the master server sends a commit message to all of the
slave servers.

62. (Currently Amended):   A computer readable <u>storage</u> medium comprising code to:
at a master server, <u>without being requested by the slave server</u> send a first delta update to
a slave server, the first delta update adapted to be used to update data from a first version to a
current version;
if data at the slave server was at the first version, receive a message that the slaver server
can commit the first delta update;
if not, receive a message from the slave server that the data at the slave server is at a
second version and, at the master server, send a second delta update adapted to be used to update
data from the second version to a current version.

63. (Currently Amended):   A computer readable <u>storage</u> medium of claim 62 containing code
to, further comprising:

storing an original copy of the data on the master server.

64. (Currently Amended):   A computer readable <u>storage</u> medium of claim 62 containing code, further comprising:

persistently caching the data on a local disk for each slave server.

65. (Currently Amended):   A computer readable <u>storage</u> medium of claim 62 containing code, further comprising:

determining a unique version number for the current state of the data on the master server if the data has changed.

66. (Currently Amended):   A computer readable <u>storage</u> medium of claim 62 containing code, wherein if the first delta update is to be committed, the first delta update is committed in a transactional manner.

67. (Currently Amended):   A computer readable <u>storage</u> medium of claim 66 containing code, wherein the slave server sends a message to the master server when it is ready to commit the first delta update.